

Système d'exploitation

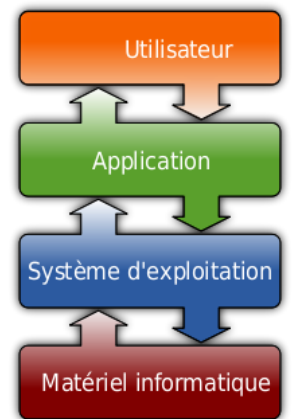
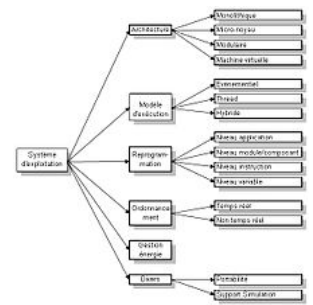
En informatique, un **système d'exploitation** (souvent appelé **OS** — de l'anglais *Operating System*) est un ensemble de programmes qui dirige l'utilisation des ressources d'un ordinateur par des logiciels applicatifs¹. Il reçoit des demandes d'utilisation des ressources de l'ordinateur — ressources de stockage des mémoires (par exemple des accès à la mémoire vive, aux disques durs), ressources de calcul du processeur central, ressources de communication vers des périphériques (pour parfois demander des ressources de calcul au GPU par exemple ou tout autre carte d'extension) ou via le réseau — de la part des logiciels applicatifs. Le système d'exploitation gère les demandes ainsi que les ressources nécessaires, évitant les interférences entre les logiciels¹.

Le système d'exploitation est un logiciel, le deuxième et le principal programme exécuté lors de la mise en marche de l'ordinateur², le premier étant le programme d'amorçage (appelé également bootloader). Il offre une suite de services généraux facilitant la création de logiciels applicatifs et sert d'intermédiaire entre ces logiciels et le matériel informatique¹. Un système d'exploitation apporte commodité, efficacité et capacité d'évolution, permettant d'introduire de nouvelles fonctions et du nouveau matériel sans remettre en cause les logiciels².

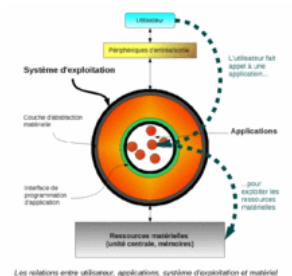
Il existe sur le marché des dizaines de systèmes d'exploitation différents², très souvent livrés avec l'appareil informatique³. C'est le cas de Windows, Mac OS, Irix, Symbian OS, GNU/Linux, (pour ce dernier il existe de nombreuses distributions) ou Android. Les fonctionnalités offertes diffèrent d'un système à l'autre et sont typiquement en rapport avec l'exécution des programmes, l'utilisation de la mémoire centrale ou des périphériques, la manipulation des systèmes de fichiers, la communication, ou la détection et la gestion d'erreurs².

La définition des systèmes d'exploitation est rendue floue par le fait que les vendeurs de ces produits considèrent le système d'exploitation comme étant la totalité du contenu de leur produit, y compris les vidéos, les images et les logiciels applicatifs l'accompagnant¹, sans compter les modifications des constructeurs informatiques (dans les cas de licence OEM) qui compliquent la donne.

En 2012, les deux familles de systèmes d'exploitation les plus populaires sont Unix (dont macOS, GNU/Linux, iOS et Android) et Windows. Cette dernière détient un quasi-monopole sur les ordinateurs personnels avec près de 90 % de part de marché depuis 5 ans⁴.



Le système d'exploitation est un intermédiaire entre les logiciels d'application et le matériel.



Relations dans un système d'exploitation moderne.

Sommaire

Fonctionnalités

- Utilisation
- Processeur
- La communication inter-processus
- Mémoire
 - La mémoire virtuelle
- Périphériques
- Fichiers
- Réseau
- Contrôle d'accès
- Interface utilisateur
 - L'interface utilisateur graphique
- Logiciels utilitaires

Typologie

- Première génération : Traitement par lots
- Deuxième génération : la multiprogrammation
- Troisième génération : le temps partagé
- Quatrième génération : le temps réel
- Cinquième génération : les systèmes distribués

Histoire

- 1960, la multiprogrammation
- 1972, les micro-ordinateurs
- 1990, les systèmes d'exploitation sous licence libre

Organisation générale du noyau

- Noyau monolithique
- Noyau à micronoyau
- Noyau hybride
- Système organisé par couches
- Appels système
- Noyau client-serveur
- Principe de machine virtuelle

Quelques exemples

Le marché

- Choix par l'acheteur
- Concurrence, compatibilité et interopérabilité
 - La guerre des Unix et l'Open Group
- Microsoft et la concurrence
- Les réseaux informatiques

Notes et références

- Notes
- Références
- A.Tanenbaum, *Systèmes d'exploitation*

Bibliographie

Annexes

- Articles connexes
- Liens externes

Fonctionnalités

Le système d'exploitation offre une suite de services généraux facilitant la création et l'utilisation de logiciels applicatifs. Les services offerts sont en rapport avec l'utilisation des ressources de l'ordinateur par les programmes². Ils permettent en particulier d'exécuter des programmes, de lire et écrire des informations, de manipuler les fichiers, de communiquer entre ordinateurs et de déceler des erreurs². Ces services permettent à plusieurs usagers et plusieurs programmes de se partager les ressources de l'ordinateur⁵. Le principal rôle du système d'exploitation est alors de gommer les différences entre les différentes architectures informatiques, et d'organiser l'utilisation des ressources de façon rationnelle :

- Utilisation des périphériques : chaque périphérique a ses propres instructions, avec lesquelles il peut être manipulé. Le système d'exploitation en tient compte. Il permet au programmeur de manipuler le périphérique par de simples demandes de lecture ou d'écriture, lui évitant la perte de temps de traduire les opérations en instructions propres à un périphérique⁵.
- Accès aux fichiers : en plus des instructions propres à chaque périphérique (décodeur de disquette, disque dur, lecteur de CD-ROM), le système d'exploitation tient compte du format propre de chaque support servant au stockage de fichiers. Il définit également des mécanismes de protection permettant de contrôler quel utilisateur peut manipuler quel fichier⁵.
- L'accès aux ressources : Une des fonctions du système d'exploitation est de protéger les ressources contre l'utilisation par des personnes non autorisées, et de résoudre les conflits lorsque deux utilisateurs demandent simultanément la même ressource⁵.
- Détection et récupération en cas d'erreur : lorsqu'une erreur survient, qu'elle soit matérielle ou logicielle, le système d'exploitation traite l'erreur en adoucissant son impact sur le système informatique. Il peut tenter de réitérer l'opération, arrêter l'exécution du programme fautif, ou signaler le problème à l'utilisateur⁵.
- Contrôle : un système d'exploitation peut tenir des statistiques d'utilisation des ressources, surveiller la performance, et les temps de réponse⁵.

La palette des services offerts et la manière de s'en servir diffère d'un système d'exploitation à l'autre. Le standard industriel POSIX du IEEE définit une suite d'appels systèmes standard. Un logiciel applicatif qui effectue des appels système selon POSIX pourra être utilisé sur tous les systèmes d'exploitation conformes à ce standard⁶.

Le système d'exploitation assure la réservation des différentes ressources pour les besoins des programmes exécutés simultanément. Les réservations peuvent être inscrites dans des journaux d'activités des fins de statistiques ou de dépannage et le système d'exploitation peut refuser une réservation à un utilisateur n'ayant pas reçu d'autorisation préalable².

Le matériel informatique peut exécuter des instructions, celles-ci sont rarement plus que des copies ou des additions. La traduction d'opérations complexes en suite d'instructions est une tâche fastidieuse qui incombe au système d'exploitation⁵. Le système d'exploitation prend en charge toute la manipulation du matériel, le logiciel applicatif ne peut donc pas voir de différence entre une machine simple, rudimentaire et une machine riche et complexe : les mêmes services sont offerts dans les deux cas¹.

Le système d'exploitation facilite le travail de programmation en fournissant une suite de services pouvant être utilisés par les logiciels applicatifs¹. Du point de vue du programmeur, son logiciel applicatif s'oriente en direction du système d'exploitation et du matériel, et les programmes sont considérés comme fonctionnant sur le système d'exploitation. Un système d'exploitation peut ainsi être vu comme une machine virtuelle. L'ensemble composé du matériel et du système d'exploitation forme la « machine » qui exécute le logiciel applicatif, une machine en partie simulée par du logiciel¹.

Un système d'exploitation est composé d'une large palette de programmes. La composition exacte dépend de l'usage cible et du type d'appareil informatique auquel le système est destiné (ordinateur personnel, serveur, superordinateur ou encore système embarqué).

Utilisation

Les utilisateurs et les programmeurs peuvent demander des services au système d'exploitation par son interface de programmation ses commandes ou son interface graphique⁷.

Interface de programmation

Les appels système permettent des interactions entre un programme en cours d'exécution et le système d'exploitation. L'utilisation d'appels système ressemble a celle de fonctions ou de sous-routines ordinaires en langage C ou Pascal².

Commandes

Les commandes permettent à un utilisateur ou un programme de demander une opération au système d'exploitation. Une commande est un programme qui effectue un appel système selon la demande de l'utilisateur^{2, 7}.

Interface graphique

L'interface graphique permet une manipulation intuitive par l'intermédiaire de pictogrammes. Cette interface, qui n'est pas une partie essentielle du système d'exploitation, et qui cache tous les détails intrinsèques de celui-ci, est souvent considérée comme un complément⁷.

POSIX (acronyme de l'anglais *Portable Operating System Interface*) est une norme relative à l'interface de programmation du système d'exploitation. De nombreux systèmes d'exploitation sont conformes à cette norme, notamment les membres de la famille Unix.

Processeur

Lorsqu'il est multitâche, le système d'exploitation permet à plusieurs utilisateurs de se servir de l'ordinateur et donne à chaque utilisateur l'impression qu'il est le seul à utiliser l'ordinateur⁵. Pour ce faire, l'utilisation du processeur est planifiée : chaque programme est exécuté durant une tranche de temps déterminé, puis le système d'exploitation bascule sur l'exécution d'un autre programme⁵.

Processus

Un processus est un ensemble d'instruction qui sont en train d'être exécutées. Les instructions proviennent d'un programme, et l'exécution nécessite du temps, de la mémoire, des fichiers et des périphériques⁸. Le système d'exploitation s'occupe de créer d'interrompre, et de supprimer des processus. Plusieurs processus se trouvent en mémoire centrale en même temps⁷.

La responsabilité du système d'exploitation et de réserver de la mémoire, et de planifier l'exécution, de s'occuper des interblocages⁹ et d'assurer les communications entre les processus⁸. L'ordonnanceur (anglais *scheduler*) associe un processus à un processeur puis plus tard le dissocie du processeur pour associer un autre processus. cette opération associer / dissocier est appelée context switch⁷. Lors de la planification, le système d'exploitation tient compte de la disponibilité, ou non, des ressources utilisées par le processus⁹. Certains systèmes d'exploitation créent des processus pour effectuer certaines tâches propre au système⁶.

Lors de l'exécution du programme, celui-ci peut demander de réserver des ressources de l'ordinateur⁶. L'exécution de ce programme peut être interrompue et le système d'exploitation continue l'exécution d'un autre programme, cependant les ressources restent réservées⁶. Lorsqu'un autre processus demande une ressource déjà réservée, le processus est mis en attente. En situation de forte compétition plusieurs processus peuvent être en attente de la même ressource⁶. Lorsque la ressource est libérée le système d'exploitation décide alors quel est le prochain processus auquel la ressource sera réservée. La stratégie d'octroi des ressources par le système d'exploitation vise à répondre équitablement à toutes les demandes et éviter des conflits⁸.

La communication inter-processus

Dans certains logiciels applicatifs, plusieurs programmes effectuent la même tâche simultanément, et s'échangent des informations⁶. Le mécanisme de protection de la mémoire (voir plus loin) empêche aux programmes de manipuler les mêmes informations, et ceux-ci doivent faire appels à des services du système d'exploitation⁶.

Protection

Par mesure de sécurité, le système d'exploitation réserve à chaque programme un *espace d'adressage* - un emplacement en mémoire que seul le programme en question peut manipuler⁶. Le système d'exploitation détecte toute tentative d'accès en dehors de l'espace d'adressage et provoque l'arrêt immédiat du programme qui tente d'effectuer telles opérations⁶. On nomme cela une erreur de protection générale.

Mémoire

Le système d'exploitation dirige l'utilisation de la mémoire. Il retient la liste des emplacements de mémoire utilisés, et par qui, ainsi que la liste des emplacements libres¹⁰. Le système d'exploitation réserve un emplacement de mémoire lorsqu'un processus le demande, et le libère lorsqu'il n'est plus utilisé, par exemple lorsque le processus s'est arrêté¹⁰.

Les fonctions de contrôle de l'utilisation de la mémoire vont en particulier suivre l'utilisation de celle-ci, quels emplacement sont libres, sont utilisés, et par qui². Ces fonctions vont également décider quel programme reçoit de la mémoire, quand et quelle quantité est mise à disposition et récupérer la mémoire qui était utilisée par un programme après son exécution, que le programme se soit arrêté volontairement ou accidentellement².

La quantité de mémoire utilisée par l'ensemble du système informatique dépend essentiellement de la manière dont le système d'exploitation effectue les réservations⁸.

Dans les systèmes d'exploitation contemporains, plusieurs programmes sont exécutés en même temps et utilisent simultanément la mémoire centrale⁶. Si un processus modifie - accidentellement ou intentionnellement - un emplacement de mémoire utilisée par un autre processus, il met celui-ci en danger⁶. S'il modifie un emplacement utilisé par le système d'exploitation il met en danger l'ensemble du système informatique⁶.

Pour éviter tel incident, le système d'exploitation réserve à chaque programme un *espace d'adressage* - un emplacement en mémoire que seul le programme en question peut manipuler⁶. Le système d'exploitation détecte toute tentative d'accès en dehors de l'espace d'adressage et provoque l'arrêt immédiat du programme qui tente d'effectuer telles opérations⁶, par le biais d'une erreur de protection générale.

La mémoire virtuelle

Le mécanisme dit de *mémoire virtuelle* est destiné à simuler la présence ou l'absence de mémoire centrale par manipulation de l'unité de gestion mémoire — un composant électronique (anglais *Memory Management Unit* abrégé *MMU*). C'est un mécanisme courant dans les systèmes d'exploitation contemporains.

La mémoire virtuelle permet d'exécuter simultanément plus de programmes que ce que la mémoire centrale peut contenir. Chaque programme n'ayant pas besoin que la totalité des informations qu'il manipule soit présente dans la mémoire centrale^{N 1}, une partie des informations est stockée dans la mémoire de masse (en général dans un fichier ou une partition de disque dur) habituellement plus importante mais plus lente et sont transférées en mémoire centrale lorsque le programme en a besoin^{T 1}.

Les programmes disposent d'un (ou plusieurs) espaces virtuels de mémoire continus pour travailler. Les adresses des données sont dites *virtuelles* dans la mesure où l'information adressée ne se trouve pas forcément ni en mémoire centrale, ni à l'adresse indiquée. Lorsque le programme essaie de lire ou écrire une donnée dans sa mémoire virtuelle, l'unité de gestion de mémoire cherche l'adresse physique correspondant à l'adresse virtuelle sollicitée grâce à une table de correspondance. Si l'emplacement n'est pas présent en mémoire centrale (on appelle cela une *faute de page*), il n'y aura évidemment aucune adresse physique correspondante. Le système d'exploitation devra alors chercher à libérer un espace en mémoire centrale en échangeant (anglais *swap*) le contenu d'un emplacement donné de mémoire centrale avec le contenu sollicité, qui se trouve en mémoire de masse^{N 2, T 2}. Cette opération s'effectue automatiquement, à l'insu des programmes.

Des mémoires associatives, incorporées dans l'unité de gestion de mémoire, accélèrent le calcul des adresses^{T 3}. Les systèmes d'exploitation utilisent généralement deux mémoires associatives : une pour le *mode noyau* et une pour le *mode utilisateur*. La mémoire du *mode noyau* est arrangée de manière à permettre au processeur d'utiliser la totalité de la mémoire centrale disponible — lors de l'exécution des programmes du noyau du système d'exploitation. Tandis que celle du mode utilisateur est arrangée de manière à protéger le noyau (qui est ainsi invisible pour le programme en question) — lors de l'exécution des programmes hors du noyau. C'est ce que l'on nomme la protection, et ces mécanismes constituent les principales caractéristiques mode protégé.

Chaque programme dispose de sa propre table de correspondance, ce qui permet de les isoler les uns des autres. Lors d'une commutation de contexte, le système d'exploitation placera la table du programme courant dans la mémoire associative^{T 4}. Le système d'exploitation crée également de nouvelles tables pour les programmes qui démarrent et décide quels emplacements de mémoire virtuelle seront ou ne seront pas présents en mémoire centrale^{T 5}.

Périphériques

Les périphériques sont tous les dispositifs informatiques qui permettent au processeur de communiquer avec l'extérieur: clavier, imprimante, carte réseau, mémoire, disque dur⁶. Ils permettent en particulier de recevoir des informations, d'en envoyer ainsi que de stocker des informations - les collecter dans le but de les renvoyer plus tard⁶.

Une des responsabilités du système d'exploitation est de suivre l'état d'utilisation - libre ou réservé - de tout le matériel du système informatique. Lorsqu'un matériel libre est demandé par un processus, il est alors réservé à ce processus⁹. Pour utiliser un périphérique, le système d'exploitation se sert d'un contrôleur et d'un pilote de périphérique.

Contrôleur

Un contrôleur est un composant électronique, qui comporte une mémoire tampon, et manipule un certain type de périphérique (disque dur, imprimante, mémoire, lecteur de bande magnétique...) ⁹. Le contrôleur est souvent intégré au périphérique⁶. Les différents contrôleurs disponibles sur le marché ne s'utilisent pas tous de la même manière⁶.

Pilote (anglais *driver*)

Les instructions de manipulation d'une gamme de contrôleurs donnée sont incluses dans un pilote informatique : un logiciel qui exploite les possibilités offertes par les contrôleurs⁶. Les pilotes informatiques font partie du système d'exploitation, et offrent des services uniformes utilisés par les autres programmes du système d'exploitation⁶.

Il existe deux stratégies de manipulation des contrôleurs⁶ : Les contrôleurs rapides sont manipulés en *programmed I/O*, dans cette stratégie le processeur envoie des demandes d'opérations au contrôleur puis vérifie de manière intermittente l'état du contrôleur pour vérifier si l'opération demandée est terminée⁶.

Les contrôleurs moins rapides sont manipulés en *interrupt driven I/O*⁶. Dans cette stratégie le processeur effectue une demande au contrôleur, puis continue d'exécuter des logiciels applicatifs. Le contrôleur envoie un signal électrique lorsque l'opération est terminée. Lors de la venue de ce signal, le processeur interrompt l'exécution des logiciels applicatifs et exécute un programme particulier *interrupt service routine* qui vérifie le nouvel état du contrôleur⁶.

Certains périphériques ne peuvent pas être partagés, et leur utilisation est alors dédiée à un seul programme à la fois. Certains périphériques peuvent être virtuels, ou leur utilisation peut être indirecte. Par exemple l'utilisation d'une imprimante n'entraîne pas une impression immédiate parce que les informations sont tout d'abord mises en attente. Cette technique du spool permet l'utilisation partagée d'un périphérique qui sans ça ne pourrait pas être partagée².

Fichiers

Un fichier est une collection d'informations portant un nom, enregistrée sur un média tel qu'un disque dur, une bande magnétique ou un disque optique⁸. Chaque médium a ses propres caractéristiques et sa propre organisation⁸.

Le système d'exploitation s'occupe de créer et de détruire des fichiers et des répertoires, de réserver de l'espace sur les médias ainsi que copier le contenu des fichiers de et vers la mémoire centrale⁸. Il aide également les logiciels applicatifs à : retrouver les fichiers, partager les fichiers entre plusieurs utilisateurs, modifier le contenu des fichiers, créer des répertoires pour permettre de classer et organiser les fichiers⁹. La vitesse du système informatique dépendra de la vitesse de manipulation des fichiers⁸.

Le système d'exploitation permet en particulier de manipuler les attributs: les caractéristiques du fichier tels que son nom, la date de création, le type du contenu, la taille et l'emplacement⁹. Il permet également de manipuler les permissions: des autorisations qui indiquent si un utilisateur pourra lire, écrire ou exécuter le fichier⁹.

Le système d'exploitation tient compte du système de fichiers: la manière dont les fichiers sont organisés et répartis sur un dispositif de stockage⁹.

Les mémoires amovibles telles que les CD ou les DVD ont une disposition normalisée dont les spécifications sont publiques, ce qui assure leur reconnaissance par tous les systèmes d'exploitation (voir ISO 9660 et UDF ainsi que l'hybride ISO/UDF).

Lorsque le système de fichier est distribué, et que les fichiers sont donc stockés sur différents ordinateurs d'un réseau informatique, le système d'exploitation envoie une requête à l'ordinateur stockant le fichier pour chaque opération à effectuer (voir NFS ou CIFS).

Dans un système d'exploitation multi-utilisateurs, les programmes manipulant le système de fichiers effectuent des contrôles pour vérifier qu'aucun fichier n'est manipulé par une personne non autorisée. Ce type de système d'exploitation refusera toute manipulation non autorisée.

Réseau

Dans un réseau informatique, deux ordinateurs reliés communiquent dès lors que les communications se font de part et d'autre selon les mêmes protocoles réseau. Selon le modèle OSI, les différents protocoles existants sont répartis sur sept niveaux, où un protocole d'un niveau donné peut être combiné avec n'importe quel protocole des niveaux situés en dessus et en dessous (encapsulation).

Un système d'exploitation contient typiquement plusieurs programmes nécessaires pour des échanges d'informations dans différents protocoles des niveaux 1 à 4. Tandis que les niveaux 5 à 7 sont pris en charge par les logiciels applicatifs et les middleware.

Pour les échanges d'informations selon les protocoles de niveau 1 et 2, le système d'exploitation demande l'opération au matériel de l'ordinateur par l'intermédiaire d'un pilote informatique, pilote qui peut faire partie intégrante du système d'exploitation ou être fourni par le constructeur du matériel.

Lors de l'envoi d'informations sur le réseau, un logiciel applicatif crée une information, la met en forme conformément aux protocoles des niveaux 7 à 5, puis la transmet au système d'exploitation. Divers programmes du système d'exploitation vont découper cette information en trames, puis vont mettre en forme les trames et les envoyer conformément aux protocoles des niveaux 4 à 1.

Lors de la réception de trames depuis le réseau, divers programmes du système d'exploitation vont tenter de les décoder conformément à différents protocoles des niveaux 1 à 4, puis transformer la suite de trames en un flux continu, qui sera envoyé au logiciel applicatif destinataire. Le logiciel va alors décoder le flux conformément aux protocoles de niveaux 5 à 7. Le logiciel applicatif effectue préalablement une *connexion*, c'est-à-dire une liaison logique par laquelle il va s'associer avec un flux particulier

Le choix exact des protocoles utilisés dépend de l'ordinateur concerné et des liaisons réseau qui vont être utilisées. Divers paramètres de configuration permettent d'influencer le choix des protocoles. Ils permettent par exemple d'empêcher l'utilisation de protocoles interdits sur le réseau concerné.

Contrôle d'accès

Les systèmes d'exploitation contemporains permettent à plusieurs usagers de se servir simultanément de l'ordinateur⁸. Le système d'exploitation comporte des mécanismes destinés à contrôler l'utilisation des ressources par les utilisateurs, les processus et les programmes⁸. Ces mécanismes permettent de certifier l'identité du programme ou de l'utilisateur et lui autoriser l'accès à une ressource en application de règlements de sécurité⁸.

Les mécanismes de sécurité du système d'exploitation servent à protéger le système informatique de l'intérieur comme de l'extérieur: Les mécanismes de sécurité intérieure protègent les processus l'un de l'autre, et assurent la fiabilité du système informatique⁹. Les mécanismes de sécurité extérieure protègent les données et les programmes enregistrés dans l'ordinateur contre des accès non autorisés et des erreurs de manipulation⁹. Le système d'exploitation empêche la lecture par des personnes non autorisées, la falsification, la suppression de données, ainsi que l'utilisation non autorisée de périphériques⁹.

Plusieurs programmes sont exécutés en même temps et utilisent simultanément la mémoire centrale⁶. Si un processus modifie - accidentellement ou intentionnellement - un emplacement de mémoire utilisée par un autre processus, il met celui-ci en danger⁶. S'il modifie un emplacement utilisé par le système d'exploitation il met en danger l'ensemble du système informatique⁶

Pour éviter tel incident, le système d'exploitation réserve à chaque programme un *espace d'adressage* - un emplacement en mémoire que seul le programme en question peut manipuler⁶. Le système d'exploitation détecte toute tentative d'accès en dehors de l'espace d'adressage et provoque l'arrêt immédiat du programme qui tente d'effectuer telles opérations⁶.

Le système d'exploitation va également refuser la mise hors service de programmes centraux tels que les logiciels serveur ou des programmes du système d'exploitation par tout utilisateur qui n'a pas préalablement reçu le privilege d'effectuer cette opération — selon les règlements introduits par l'administrateur de sécurité.

Lorsqu'un logiciel autonome (bot informatique) demande des opérations au système d'exploitation, le logiciel doit préalablement décliner son *identité en tant que produit* puis, sur la base de cette identité, le système d'exploitation effectue les mêmes vérifications que pour une personne physique.

Les mécanismes de contrôle d'accès ont aussi pour effet de lutter contre les logiciels malveillants — ceux-ci effectuent souvent des opérations susceptibles de perturber l'utilisation de l'ordinateur

Interface utilisateur

Un ensemble de programmes du système d'exploitation reçoit les informations envoyées par les logiciels applicatifs, et les place sur une image numérique qui sera envoyée au matériel par l'intermédiaire d'un pilote. En complément un autre ensemble de programmes reçoit les manipulations effectuées par l'utilisateur par l'intermédiaire de pilotes puis les transmettent au logiciel concerné. Ces deux ensembles créent l'interface homme-machine qui permet à un usager de dialoguer avec la machine.

Le système d'exploitation peut dialoguer avec un usager par l'intermédiaire d'un autre ordinateur ou d'un terminal (interface *distribuée*). Les informations envoyées par les logiciels applicatifs seront alors envoyées à l'autre ordinateur selon un protocole prévu à cet effet, tandis que l'autre ordinateur enverra les manipulations effectuées par l'utilisateur. Voir SSH, RFB ou X Window System.

Lorsque l'interface est en mode texte, l'image numérique est une grille dans laquelle sont placés des caractères d'imprimerie, la grille comporte typiquement 80 colonnes et 35 lignes. L'interface se manipule avec un clavier. Ce type d'interface, qui existe depuis les débuts de l'informatique^{N 3} est aujourd'hui remplacé par les interfaces graphiques.

L'interface utilisateur graphique

Dans une interface utilisateur graphique (anglais *Graphical User Interface* abrégé *GUI*), l'image numérique est composée par un programme du système d'exploitation par superposition de points, de lignes, de pictogrammes et de caractères d'imprimerie. L'interface se manipule typiquement avec une souris selon le principe WIMP (anglais *Windows, Icons, Menus and Pointer device*). L'image numérique est créée à l'aide du processeur graphique de l'ordinateur.

Lors des manipulations de la souris, le système d'exploitation déplace l'élément d'image qu'est le pointeur et effectue les calculs nécessaires pour déterminer quel est l'élément de l'image qui se trouve juste en dessous. À chaque élément de l'image peut être associé un programme. Un widget est un programme qui dessine et anime un élément d'image dont l'aspect peut-être celui d'un bouton poussoir, d'une lampe témoin, d'un ascenseur d'une zone texte, d'un menu, etc. Divers widgets sont fournis avec le système d'exploitation.

Les programmes pour interface graphique sont aujourd'hui (2011) inclus dans tous les systèmes d'exploitation contemporains. Le X Window System est l'ensemble des programmes pour interface utilisateur graphique inclus dans tous les systèmes d'exploitation de la famille Unix. Pour Windows, l'équivalent est le programme Explorer, aussi nommé Explorateur Windows (à ne pas confondre avec Internet Explorer).

Logiciels utilitaires

Un logiciel applicatif sert à assister l'utilisateur dans une activité. Les logiciels utilitaires sont des logiciels applicatifs qui permettent à l'utilisateur d'effectuer des manipulations basiques telles que démarrer des programmes, copier des fichiers ou modifier des paramètres de configuration. Divers logiciels utilitaires sont fournis avec les systèmes d'exploitation.

Un interpréteur de commandes est un programme qui permet d'exécuter d'autres programmes en écrivant leur nom éventuellement suivi de divers paramètres. Il est accompagné de plusieurs programmes qui permettent la manipulation des fichiers (copie, changement de nom...). Ce type de programme est utilisé pour effectuer des manipulations ou exécuter des scripts — suites de manipulations pré-enregistrées (voir commande informatique). Ce type de programme est souvent fourni avec le système d'exploitation, mais rien n'y oblige, et on peut très bien en importer.

Un environnement de bureau est un programme dans lequel les différents éléments de l'ordinateur (programmes, fichiers, disques durs) sont présentés sous forme de pictogrammes sur lesquels il est possible d'effectuer différentes actions. Il permet d'exécuter des programmes, d'effectuer différentes opérations sur les fichiers (copie, changement du nom, déplacement ou suppression).

Certains programmes permettent à l'utilisateur de modifier les paramètres de configuration du système d'exploitation. Ceux-ci proposent des listes à choix multiples et effectuent des contrôles de validité avant de modifier les paramètres.

D'autres programmes servent à installer des logiciels, c'est-à-dire copier les fichiers dans les emplacements prévus à cet effet, et effectuer les modifications de configuration nécessaire pour rendre le logiciel opérationnel. Ces programmes peuvent aussi servir à consulter la liste des logiciels actuellement installés dans l'ordinateur

Un système d'exploitation multi-utilisateurs est en général fourni avec des programmes permettant de surveiller l'utilisation — par autrui — de l'ordinateur — consultation de journaux d'activité — ou de modifier les listes dedroits d'accès en vue d'autoriser ou d'interdire un fichier à certains utilisateurs.

Typologie

Il existe cinq générations de systèmes d'exploitation *par lots* (*batch*), multi programmés, en temps partagé, temps réel, et distribués. Chacun des principes mis en œuvre dans une génération se retrouve dans les générations suivantes⁵.

- Un système de traitement par lots (*batch*) est prévu pour l'exécution de grands calculs les uns après les autres, avec peu d'intervention utilisateur²

À partir de la génération des systèmes d'exploitation multi-programmés, plusieurs programmes sont exécutés simultanément par planification (*scheduling*). Dans ces systèmes d'exploitation multitâches, plusieurs programmes résident dans la mémoire centrale et le système d'exploitation suspend régulièrement l'exécution d'un programme pour continuer l'exécution d'un autre⁵

- Dans la génération des systèmes multi-programmés, l'exécution simultanée de plusieurs programmes vise l'utilisation efficace de la puissance de calcul du processeur
- Dans la génération des systèmes en temps partagé l'exécution simultanée de plusieurs programmes vise à répondre rapidement aux demandes de plusieurs utilisateurs en communication directe avec l'ordinateur⁵.
- Un système d'exploitation temps réel doit garantir que toute opération se termine dans un délai donné, en vue de garantir la réussite du dispositif dans lequel l'ordinateur est utilisé⁵.
- Un système distribué dirige l'utilisation des ressources de plusieurs ordinateurs à la fois. Il utilise les capacités d'un réseau informatique, contrôle un groupe de machines, et lui fait apparaître comme une machine unique, virtuelle, de très grande capacité

Chaque système d'exploitation est conçu pour fonctionner avec une gamme particulière de machines (type de processeur, constructeur, architecture). Si un système d'exploitation est disponible pour plusieurs gammes de machines différentes, alors le même code source est compilé^{N4} et adapté à chaque gamme de machines. La palette de pilotes inclus dans le système d'exploitation est adaptée au matériel informatique disponible sur le marché pour cette gamme de machines.

Première génération : Traitement par lots

Les systèmes d'exploitation basés sur le traitement par « lots » (suites d'instructions et de données dans un ensemble de cartes perforées) sont apparus dans les années 1950. Un programme (avec ses données) n'est rien d'autre qu'une pile de cartes avec des indicateurs de début et de fin de lot. L'exécution d'un programme consiste à demander à un opérateur de placer la pile de cartes dans le lecteur, puis l'opérateur lance la lecture séquentielle des cartes. Le processeur central est au repos, durant les manipulations de l'opérateur⁵

Un *batch* est un lot de travaux à effectuer. L'opérateur compose un *batch* en posant les unes sur les autres les piles de cartes des différents programmes (avec leurs données) demandés par les utilisateurs. Il forme une grande pile de cartes séparées par des marque-page, en général une carte de couleur particulière, qu'il place ensuite dans le lecteur. Le regroupement de plusieurs programmes en un *batch* diminue les interventions de l'opérateur⁵.

Dans un système basé sur les *batches*, le cœur du système d'exploitation est un programme moniteur qui réside continuellement en mémoire centrale et permet à l'opérateur de demander le début ou l'arrêt de l'exécution du lot. À la fin de l'exécution de chaque tâche du lot, le moniteur effectue des travaux de nettoyage, puis lance l'exécution de la tâche suivante. Ainsi, l'opérateur intervient uniquement au début et à la fin du lot⁵.

Dans ces systèmes d'exploitation les commandes ajoutées au marque-page, formulées dans le langage JCL (*Job Control Language*) sont un des seuls moyens qu'a l'utilisateur d'interagir avec le système d'exploitation⁵.

Les systèmes d'exploitation *batch* sont adaptés à des applications nécessitant de très gros calculs mais peu d'implication de l'utilisateur : météo, statistiques, impôts... Les utilisateurs n'attendent pas immédiatement de résultats. Ils soumettent les demandes, puis reviennent ultérieurement collecter les résultats⁵

En raison de la grande différence de vitesse entre le processeur et les périphériques, dans un système d'exploitation *batch* le processeur est inutilisé 90 % du temps car les programmes attendent qu'un périphérique ou un autre termine les opérations. Avec ces systèmes d'exploitation il n'y a pas de concurrence entre les différentes tâches, la mise en œuvre de l'utilisation du processeur, de la mémoire et des périphériques est triviale⁵ mais loin d'être optimale.

Deuxième génération : la multiprogrammation

Les systèmes d'exploitation multi-programmés sont apparus dans les années 1960. Le but recherché par de tels systèmes est d'augmenter l'efficacité de l'utilisation du processeur et des périphériques en utilisant la possibilité de les faire fonctionner en parallèle. Plusieurs programmes sont placés en mémoire centrale, et lorsque le programme en cours d'exécution attend un résultat de la part d'un périphérique, le système d'exploitation ordonne au processeur d'exécuter un autre programme⁵

Dans les systèmes d'exploitation multi-programmés, l'utilisation du processeur est partagée par planification (*scheduling*) : à chaque utilisation d'un périphérique, le système d'exploitation choisit quel programme va être exécuté. Ce choix se fait sur la base de priorités. Le système d'exploitation comporte un mécanisme de protection évitant ainsi que le programme en cours d'exécution ne lise ou n'écrive dans la mémoire attribuée à un autre programme. Les programmes sont exécutés dans un *mode non-privilégié*, dans lequel l'exécution de certaines instructions est interdite⁵.

Les systèmes multi-programmés nécessitent un ordinateur et des périphériques mettant en œuvre la technique du DMA (*direct memory access*) celle-ci, le processeur ordonne à un périphérique d'effectuer une opération, le résultat de l'opération est placé en mémoire centrale par le périphérique tandis que le processeur exécute d'autres instructions. Dans les systèmes multi-programmés, tout comme pour les systèmes *batch*, l'utilisateur n'a que peu de contact avec les programmes et de maigres possibilités d'intervention⁵

Troisième génération : le temps partagé

Les systèmes d'exploitation en temps partagé sont apparus dans les années 1970. Ils sont utilisés dans des dispositifs interactifs où plusieurs utilisateurs sont simultanément en dialogue avec l'ordinateur. Un système d'exploitation *en temps partagé* est destiné à répondre rapidement aux demandes de l'utilisateur et donner à chaque utilisateur l'impression qu'il est le seul à utiliser l'ordinateur⁵.

Un système en temps partagé met en œuvre des techniques sophistiquées de multiprogrammation en vue de permettre l'utilisation interactive de l'ordinateur par plusieurs usagers et plusieurs programmes simultanément². L'arrivée, en 1970, de cette nouvelle génération de systèmes d'exploitation résulte d'une forte demande des consommateurs, et de la baisse du prix du matériel informatique ayant rendu possible sa réalisation⁵.

Dans les systèmes d'exploitation en temps partagé la notion de *batch* n'a que peu d'importance. Ces systèmes mettent en œuvre de nouveaux mécanismes d'utilisation du processeur et de la mémoire, qui leur permet de répondre rapidement à des demandes provenant simultanément d'un grand nombre d'utilisateurs⁵

Dans ces systèmes, tout comme dans la génération précédente, l'utilisation du processeur est planifiée. Cependant, contrairement aux systèmes de la génération précédente, dans les systèmes en temps partagé chaque programme est exécuté durant une tranche de temps déterminé, puis le système d'exploitation bascule sur l'exécution d'un autre programme, ce qui évite qu'un programme monopolise l'utilisation du processeur au service d'un utilisateurentraînant des retards pour les autres usagers⁵.

Les systèmes d'exploitation en temps partagé mettent en œuvre la technique du *swap* : lorsque le programme en cours d'exécution a besoin de plus de mémoire que celle disponible, un autre programme inactif est retiré pour gagner de la place, le programme inactif est alors enregistré temporairement sur le disque dur. L'enregistrement sur disque provoque cependant une perte de temps non négligeable⁵.

En 2011 de nombreux systèmes d'exploitation sont basés sur Unix, un système en temps partagé³.

Quatrième génération : le temps réel

Les systèmes d'exploitation temps-réel sont apparus au milieu des années 1970, notamment chez Hewlett-Packard¹¹. Ils sont destinés aux dispositifs devant non seulement donner des résultats corrects, mais les donner dans un délai déterminé. Ces systèmes d'exploitation sont souvent utilisés par des ordinateurs reliés à un appareil externe (pilotes automatiques, robots industriels, applications vidéo et audio⁵) pour lequel un retard de réponse de l'ordinateur entraînerait un échec de l'appareil.

Dans ces systèmes d'exploitation, l'accent est mis sur la durée nécessaire pour effectuer chaque opération, pour répondre aux demandes rapidement en vue de satisfaire aux contraintes de temps du système dans lequel il est utilisé⁵.

Certains services offerts par ces systèmes d'exploitation sont réalisés comme des logiciels applicatifs, et sont exécutés en concurrence avec ceux-ci. Un système d'exploitation temps réel autorise un contact direct entre les logiciels applicatifs et les périphériques. Dans certains systèmes temps réel les ressources sont réservées, évitant ainsi les ralentissements que provoqueraient les réservations à la volée, et garantissant que les ressources sont continuellement disponibles⁵

Les systèmes d'exploitation temps-réel évitent d'utiliser la technique du *swap* en raison des risques de dépassement des délais⁵. RTX, Windows CE, Embedded Linux, Symbian OS, Palm OS et VxWorks sont des systèmes d'exploitation*temps réel*¹².

Cinquième génération : les systèmes distribués

La baisse des prix du matériel informatique a permis, dans les années 1990, la création de systèmes informatiques composés de plusieurs ordinateurs, et donc plusieurs processeurs, plusieurs mémoires, et de nombreux périphériques. Un système distribué permet le partage des ressources entre les ordinateurs. Un utilisateur d'un ordinateur bon marché peut se servir de ressources coûteuses existant sur un autre ordinateur⁵.

Mach, Amoeba, Andrew, Athena, et Locus sont des systèmes d'exploitation distribués. Ils ont tous été développés par des universités¹³.

Histoire

L'histoire des systèmes d'exploitation est fortement liée à celle des ordinateurs. Les premières générations d'ordinateurs, dans les années 1945 à 1955, ne comportaient pas de système d'exploitation. Dans ces ordinateurs équipés de tubes à vide, les programmes manipulaient les ressources matérielles de l'ordinateur sans passer par un intermédiaire¹⁴. L'ordinateur était utilisé par une seule personne à la fois : la tâche de l'opérateur consistait à placer des piles de cartes perforées dans le lecteur, où chaque carte comportait des instructions d'un programme ou des données¹⁴. Les ordinateurs à tube à vide de cette génération n'avaient qu'une faible puissance de calcul, ils étaient volumineux, peu commodes et peu fiables (les tubes à vide grillaient souvent)

1960, la multiprogrammation

Dans les années 1960, avec l'arrivée des circuits électroniques à semi-conducteurs, la puissance de calcul des processeurs a augmenté de manière significative¹⁴. Ce qui a permis la réalisation de systèmes d'exploitation rudimentaires: Les ordinateurs ont été équipés d'un *spooler* - file d'attente permettant d'utiliser la puissance de calcul du processeur pendant que l'opérateur introduit les cartes. L'utilisation des ressources matérielles par les programmes se faisaient alors par l'intermédiaire d'une bibliothèque logicielle¹⁴. Il a alors été possible de placer en mémoire plusieurs programmes simultanément et les exécuter simultanément; un programme *timesident monitor* résidait continuellement dans la mémoire centrale et contrôlait l'exécution des différents programmes¹⁴.

En 1965 le Massachusetts Institute of Technology se lance dans la création du premier système d'exploitation multitâche et multi-utilisateurs : Multics (pour *MULTiplexed Information and Computing Service*^{T 6}, ou service multiplexé d'information et de calcul). Sur le principe de la *multiprogrammation*, le système d'exploitation autorisait le chargement de plusieurs programmes en mémoire et gérât le passage de l'un à l'autre, mais cette fois-ci sans attendre le blocage d'un programme^{N 5}. Chaque programme était exécuté pendant une durée de quelques millisecondes, puis le système passait au suivant. Ce temps, très court, donnait l'illusion que les programmes s'exécutaient simultanément — une illusion qui existe encore avec les systèmes d'exploitation contemporains.

De plus, ces programmes pouvaient appartenir à des utilisateurs distincts, chacun ayant l'impression que la machine travaille uniquement pour lui. La possibilité pour un ordinateur de servir simultanément plusieurs personnes augmentait le retour sur investissement de l'achat de matériel très coûteux par les entreprises et les institutions. Cependant, du fait de son écriture dans un langage de programmation PL/I trop complexe pour les ordinateurs de l'époque, Multics fut un échec commercial. Il a cependant inspiré en grande partie la gamme de systèmes GCOS développés conjointement par Honeywell et Bull¹⁵.

En 1969, les ingénieurs Ken Thompson et Dennis Ritchie des laboratoires Bell rêvent d'utiliser le système d'exploitation Multics, mais le matériel pour le faire fonctionner est encore hors de prix. Thompson se lance dans l'écriture d'une version allégée de Multics pour un PDP-7 inutilisé. Le système, fonctionnel, est surnommé Unics (pour *UNiplexed Information and Computing Service*^{T 7}), puis finalement baptisé UNIX. Rapidement reprogrammé dans un langage de programmation plus approprié (le C, développé par Ritchie pour l'occasion), UNIX se révèle particulièrement simple à porter sur de nouvelles plateformes, ce qui assure son succès^{T 8, 16}.

1972, les micro-ordinateurs

Dès 1980, les circuits électroniques à transistor ont été remplacés par des circuits intégrés, plus petits, ce qui a permis de réaliser des appareils plus compacts et moins coûteux et lancé le marché des ordinateurs personnels. De nombreux concepteurs de système d'exploitation qui se sont lancés sur ce marché n'avaient pas d'expérience, ce qui a donné de nouveaux produits, fondés sur des nouvelles idées, sans héritage ou influence de ce qui se faisait jusqu'alors¹⁴. CP/M, mis sur le marché en 1974, a été le premier système d'exploitation pour micro-ordinateur, son caractère très *sympathique*, facile à aborder et commode (*user-friendly*) l'a rendu très populaire et influencé le marché des systèmes d'exploitation¹⁴.

En 1980, IBM prend contact avec Bill Gates, cofondateur de la société Microsoft, pour l'adaptation du langage BASIC à son nouveau micro-ordinateur : le Personal Computer (abrégé *PC*). IBM est également à la recherche d'un système d'exploitation, et Bill Gates leur conseille de se tourner vers CP/M. Mais Gary Kildall refuse de signer le contrat avec IBM. Bill Gates saute sur l'occasion : il rachète QDOS — un système d'exploitation *quick-and-dirty* pour les processeurs Intel 8086 — pour proposer à IBM le package DOS/BASIC. Après quelques modifications effectuées à la demande d'IBM, le système est baptisé MS-DOS^{T 9, 17}.

Xerox, une des sociétés majeures de l'époque, s'intéresse à l'optique de Steve Jobs. Elle réunit une poignée de scientifiques et d'ingénieurs dans son centre de recherche de Palo Alto et développe le premier micro-ordinateur équipé d'une interface utilisateur graphique, sur la base de thèses et d'études en ergonomie effectuées les années précédentes. Le résultat de ces recherches, le Xerox Star, ne sera jamais commercialisé. Dix ans plus tard, c'est Apple avec le Macintosh qui popularise les recherches effectuées par Xerox¹⁸.

1990, les systèmes d'exploitation sous licence libre

En 1983, Richard Stallman du Massachusetts Institute of Technology lance l'idée d'un système d'exploitation sous licence libre : GNU¹⁹. Il développe des outils de programmation, des logiciels utilitaires, et crée la GNU General Public License — un contrat de licence autorisant une utilisation sans restrictions ainsi que la publication du code source, sa modification, et sa redistribution. Le succès est immédiat, mais le système ne possède toujours pas, en 1990, de noyau libre, et les tentatives pour en développer un sont loin d'être abouties²⁰

En 1987, Andrew Tanenbaum, professeur à l'université libre d'Amsterdam crée le système d'exploitation Minix, clone d'UNIX dont le code source est destiné à illustrer son cours sur la construction des systèmes d'exploitation^{N 6}. Mais Minix, dont la vocation est pédagogique, comporte alors de trop nombreuses limitations techniques et ne permet pas une utilisation poussée^{N 7}.

En 1991, Linus Torvalds, étudiant à l'université d'Helsinki inspiré par les travaux de Tanenbaum, sort la toute première version (0.01) de son propre noyau : Linux, qui est au départ une réécriture de Minix. Linux passe sous licence GNU en 1992²¹ et il faut attendre 1994 pour voir la version 1.0^{T 10}, donnant ainsi naissance à la distribution d'un système d'exploitation entièrement libre GNU/Linux.

Un autre système d'exploitation libre apparaît à la même époque : 4.4BSD. La Berkeley Software Distribution(BSD) est la version d'UNIX développée par les étudiants et les chercheurs de l'université de Berkeley depuis 1977²². Les logiciels utilitaires, créés sous licence libre, sont vendus avec le noyau Unix d'AT&T, lui-même sous licence propriétaire. Cette double licence de BSD est à l'origine de plusieurs années de litige entre l'Université de Berkeley et AT&T²³. Les étudiants de l'université travaillent à remplacer les programmes développés par AT&T par leurs propres programmes, sous licence libre, afin de résoudre le litige. Cette situation dure jusqu'à la sortie de 4.4BSD en 1994, qui ne contient pratiquement plus de code d'AT&T^{T 11}.

C'est à la suite des initiatives et travaux de Linus Torvalds et de Richard Stallman, aidés par des milliers de bénévoles, et consécutivement aux travaux des étudiants de l'université de Berkeley que GNU/Linux et 4.4BSD sont devenus les premiers systèmes d'exploitation sous licence libre²⁴.

Organisation générale du noyau

Un système d'exploitation est essentiellement événementiel - il est exécuté lorsque quelque chose s'est passé, typiquement lors d'un appel système, une interruption matérielle ou une erreur²⁵. C'est un logiciel étendu et complexe, qui offre de nombreuses fonctions. Il est construit comme une suite de modules, chacun ayant une fonction déterminée²⁶.

Le noyau (en anglais : kernel) est la pièce centrale du système d'exploitation²⁷. C'est le second programme chargé en mémoire (juste après le bootloader) et il y reste en permanence - ses services sont utilisés continuellement²⁷.

Il réside généralement dans un emplacement protégé de mémoire vive, qui ne peut pas être modifié ni exploité par les autres programmes²⁷ (c'est-à-dire dans le cas d'un système d'exploitation en mode protégé).

C'est un composant critique: Si le kernel subit une erreur et s'arrête alors l'ordinateur cessera de fonctionner, tandis que si un autre programme s'arrêtait (par exemple un programme utilisateur) alors le système d'exploitation resterait opérationnel²⁷.

Il offre typiquement des fonctions pour créer ou détruire des processus (i.e. exécuter des programmes), diriger l'utilisation du processeur, de la mémoire et des périphériques. Il offre également les fonctions qui permettent aux programmes de communiquer entre eux et de s'aligner dans le temps (synchronisation)²⁶

Noyau monolithique

Dans une construction monolithique, le système d'exploitation est composé d'un seul programme: le noyau. Celui-ci est typiquement organisé en couches²⁸. La construction monolithique est la plus courante, appliquée à la plupart des Unix²⁷.

Noyau à micronoyau

Dans la construction micronoyau, le kernel fournit les services minimum²⁷: de nombreuses fonctions du système d'exploitation ont été retirées du noyau et sont offertes par des programmes manipulés par celui-ci, qui sont ce que l'on appelle des services²⁸ (pour un système en mode protégé, la grande différence avec un système monolithique sera que ces services seront exécutés dans l'espace mémoire utilisateur et non celui du noyau).

Les appels de fonction au système d'exploitation par les programmes utilisateurs ont été remplacées par des envois de message²⁸. Dans cette construction le noyau est utilisé principalement pour planifier l'exécution de processus, et pour échanger des messages²⁸. AIX, BeOS, Mach, Hurd, MacOS X, Minix et QNX sont des systèmes d'exploitation qui utilisent cette construction²⁷.

Noyau hybride

La construction hybride ressemble à une construction microkernel, cependant certaines fonctions ont été placées dans le noyau pour des raisons d'efficacité²⁷. Windows NT, 2000 et XP (et également les Windows plus récents) ainsi que DragonFly BSD sont en construction hybride²⁷.

Système organisé par couches

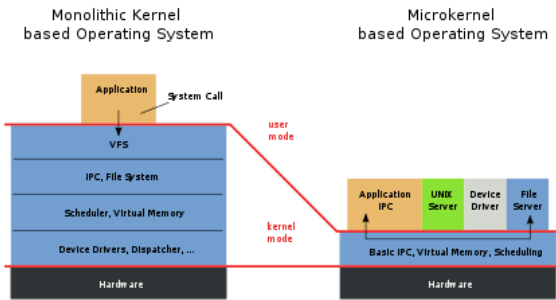
Le principe de la répartition par couches est que chaque module d'une couche donnée utilise uniquement des fonctions offertes par les modules qui se trouvent dans la couche au-dessous²⁶. L'organisation en couche n'est pas unique aux systèmes d'exploitation et est couramment appliquée aux logiciels applicatifs²⁸. Dans un système d'exploitation, les couches inférieures concernent les périphériques et la mémoire ; au-dessus desquelles se trouvent les systèmes de fichiers, puis les processus²⁸. Ces couches constituent la normalisation de la machine : pour un programme utilisateur et un système d'exploitation donné, tous les ordinateurs seront identiques (ou presque). C'est ce que l'on nomme l'abstraction matérielle²⁹

Appels système

Le kernel étant dans un emplacement protégé, il est impossible pour un logiciel applicatif d'appeler directement ses fonctions²⁷. Un mécanisme permet aux logiciels applicatifs de demander des services au système d'exploitation²⁸. Il est typiquement mis en œuvre par une bibliothèque. Celle-ci comporte des fonctions bouchon qui consistent à placer les paramètres selon une convention, puis utiliser une instruction du processeur qui provoque la mise en pause du processus en cours et l'exécution du système d'exploitation²⁸. Les fonctions de bouchon s'utilisent comme les fonctions ordinaires d'une bibliothèque²⁸.

Noyau client-serveur

Dans une construction client-serveur, le cœur du système d'exploitation a pour seule fonction d'assurer la communication entre les modules²⁶. Le système d'exploitation est divisé en nombreux petits modules qui sont exécutés de la même manière que des logiciels applicatifs. Cette construction est bien adaptée aux systèmes d'exploitation distribués²⁶



Comparaison entre noyau monolithique ou micronoyau

Principe de machine virtuelle

Le principe des *machines virtuelles* est que le système d'exploitation crée l'illusion qu'il existe plusieurs machines, aux capacités étendues, en utilisant une seule machine aux capacités plus limitées²⁶.

Le cœur du système d'exploitation est un moniteur qui crée les illusions - *machine virtuelle* -. Les caractéristiques techniques de ces machines virtuelles sont identiques à celle de la machine réellement utilisée et celles-ci peuvent être utilisées pour exécuter un autre système d'exploitation²⁶.

Quelques exemples

Comparaison des principaux systèmes d'exploitation

Année d'apparition	Nom	Famille	Éditeur	Matériel pris en charge	Utilisation	Noyau	Graphique	Multitâche	Multi-utilisateur	Multiprocesseur	Temps réel
1973	<u>SYSMIC</u>		<u>R2E</u>	<u>Micral</u>	ordinateurs personnels, stations de travail			✓	✓		
1977	<u>VMS</u>		<u>DEC</u>	<u>VAX, DEC Alpha, Hewlett-Packard</u>	serveurs, ordinateurs centraux			✓	✓		
1978-1985	<u>CP/M</u>		<u>Digital Research</u>	<u>Amstrad CPC, Commodore 128, TRS-80</u>	ordinateurs personnels				✓ (cas de MP/M)		
1981-1990	<u>DOS</u>		<u>IBM & Microsoft</u>	<u>Compatible PC</u>	ordinateurs personnels						
1982	<u>QNX</u>		<u>Quantum Software Systems</u>	compatibles PC, MIPS, PowerPC, ARM	systèmes embarqués, automates industriels	✓	✓	✓	✓		✓
1984	<u>Mac OS</u>		<u>Apple</u>	<u>Apple Macintosh</u>	ordinateurs personnels		✓				
1985	<u>TOS</u>		<u>Atari</u>	<u>Atari ST, Eagle, Medusa, Hades, Milan, Firebee, ColdFire</u>	ordinateurs personnels		✓				
1985	<u>AmigaOS</u>		<u>Commodore</u>	<u>Commodore Amiga, PowerPC</u>	ordinateurs personnels et consoles de jeu		✓	✓			✓ ³⁰
1986	<u>AIX</u>	Unix	<u>IBM</u>	<u>PS/2, RS/6000, PowerPC</u>	ordinateurs personnels, serveurs, stations de travail, superordinateurs	✓	✓ ³¹	✓	✓		
1986	<u>Irix</u>	Unix	<u>SGI</u>	machines de SGI	stations de travail et serveurs	✓	✓	✓	✓	✓	
1986-1996	<u>NeXTSTEP</u>	Unix	<u>NeXT</u>	Compatible PC, SPARC, Hewlett-Packard	stations de travail	✓	✓	✓	✓		
1987-2006	<u>OS/2</u>		<u>IBM et Microsoft</u>	<u>PS/2 et Compatible PC</u>	ordinateurs personnels	✓	✓	✓			
1987	<u>Minix</u>		<u>Andrew Tanenbaum</u>	Compatible PC, m68k, SPARC	(pédagogique) ^{N 8}	✓	✓ ³²	✓			
1989	<u>Symbian OS</u> ^{N 9}		<u>Symbian Ltd</u> ^{N 10}	<u>Nokia, Siemens, Samsung, Panasonic</u>	téléphones mobiles, smartphone, assistants personnel	✓ ³³	✓	✓			✓
1990	<u>Windows 3.x</u>	Windows	<u>Microsoft</u>	Surcouche logicielle à DOS	ordinateurs personnels	✓ ³⁴	✓	✓			
1991	<u>Solaris</u>	Unix	<u>Sun</u>	machines de Sun et x86/64	serveurs, stations de travail, superordinateurs	✓	✓	✓	✓	✓	
1991	<u>GNU/Linux</u>	Unix	(communautaire)	nombreux ^{N 11}	tous	✓	✓	✓	✓	✓	✓ ^{N 12}
1991 ^{N 13}	<u>Windows NT</u>	Windows	<u>Microsoft</u>	Compatible PC	serveurs, stations de travail, ordinateurs personnels	✓	✓	✓	✓	✓	
1994 ^{N 14}	<u>NetBSD</u>	Unix	(communautaire)	nombreux ^{N 15}	tous	✓	✓	✓	✓	✓	
1994 ^{N 14}	<u>FreeBSD</u>	Unix	(communautaire)	nombreux ^{N 16}	tous	✓	✓	✓	✓	✓	
1994 ^{N 14}	<u>OpenBSD</u>	Unix	(communautaire)	nombreux ^{N 17}	tous	✓	✓	✓	✓	✓	
1996	<u>Windows CE</u>	Windows	<u>Microsoft</u>	<u>x86, MIPS, ARM</u> ³⁵	smartphone, assistants personnels, automates industriels	✓ ³⁶	✓	✓			✓
1996	<u>RTX</u>		<u>Intervalzero</u>	<u>x86, x64</u> ³⁷	ordinateurs industriels	✓		✓		✓	✓
1999 ^{N 18}	<u>Mac OS X</u>	Unix	<u>Apple</u>	<u>x86, PowerPC de Apple</u>	ordinateurs personnels, serveurs, station de travail	✓	✓	✓	✓	✓	✓
1999	<u>BlackBerry OS</u>		<u>Research In Motion</u>	téléphones mobiles de BlackBerry	smartphone	✓	✓	??			??
2007	<u>Android</u>	Unix	consortium Open Handset Alliance	produits des fabricants du	smartphone, tablette	✓	✓	✓	✓	??	??

				Open Handset Alliance	électronique						
2007	iOS	Unix	Apple	appareils de Apple (iPhone, iPod, iPad...)	smartphone, tablette électronique, baladeur numérique	✓	✓	✓			??
2007	Sailfish OS	Unix	Jolla	appareils de la firme Finlandaise Jolla	smartphone, tablette électronique	✓	✓	✓			??

Le marché

Née en 1985, la gamme des systèmes Windows de Microsoft équipe en 2008 près de 90 % des ordinateurs personnels, ce qui la place en situation de monopole notamment auprès du grand public. En 2008, ses parts de marché sont descendues en dessous de 90 % pour la première fois depuis 15 ans³⁹. Puis à la suite de la croissance très rapide du marché des smartphones, et du retard pris par Microsoft sur ce marché, ses parts de marché sur les appareils personnels sont passées de 95 % en 2005 à 20 % en 2013⁴⁰.

Amorcée en 1969, la famille de systèmes d'exploitation Unix compte plus de 25 membres⁴¹. GNU/Linux, BSD et Mac OS X sont aujourd'hui les systèmes d'exploitation les plus populaires de la famille Unix.

La famille Windows équipe aujourd'hui 38 % des serveurs tandis que la famille Unix équipe 31 %, dont à peu près la moitié avec GNU/Linux⁴². La famille Unix anime 60 % des sites web dans le monde⁴³ et GNU/Linux équipe 95 % des près de 500 superordinateurs du monde⁴⁴. En 2012, la famille Unix anime 90 % des smartphones [réf. nécessaire].

Né en 1990, Symbian OS est en 2007 le système d'exploitation le plus répandu sur les téléphones mobiles et assistants personnels avec 67 % de part de marché⁴⁵. En 2012, les quatre systèmes d'exploitation Android de Google, Symbian, iOS de Apple et Blackberry de Research In Motion occupent ensemble 95 % du marché des smartphones. Android, le plus populaire (75 %), est en progression, tandis que les autres sont en recul. Les parts de marché de Symbian ne sont plus que de 2,3 %⁴⁶.

Du côté des tablettes tactiles, iOS de Apple était le premier système d'exploitation largement diffusé avec plus de 80 % de part de marché en 2010⁴⁷. Trois ans plus tard sa part de marché est de 20 % et celle de Android est de plus de 56 %⁴⁸.

Les serveurs et superordinateurs sont majoritairement équipés de systèmes d'exploitation de la famille UNIX⁴⁹.

Choix par l'acheteur

De nombreux logiciels applicatifs sur le marché sont construits pour fonctionner avec un système d'exploitation en particulier, ou une famille en particulier et un système d'exploitation est construit pour fonctionner avec une gamme de machines donnée. Pour l'acheteur le choix de la famille de machine limite le choix du système d'exploitation, qui lui-même limite le choix des logiciels applicatifs.

Chaque système d'exploitation, selon la palette de programmes qu'il contient, est construit pour fonctionner avec certains réseaux informatiques. Pour l'acheteur qui possède un réseau informatique (typiquement les entreprises et les institutions) le choix du système d'exploitation dépend de son adéquation au réseau existant de l'acheteur.

L'utilité d'un système d'exploitation pour l'utilisateur est proportionnel au nombre de logiciels applicatifs qui sont prévus pour lui. La popularité élevée d'un système d'exploitation attire les éditeurs de logiciels applicatifs, ce qui accroît encore sa popularité (effet réseau). Ce phénomène fait que le marché est sujet aux situations de monopole.

Apple, Sun Microsystems et Silicon Graphics sont des marques qui fabriquent du matériel informatique et développent des systèmes d'exploitation pour leur propre matériel. Certains systèmes d'exploitation, comme Microsoft Windows et Android, sont vendus avec le matériel informatique, conformément à des accords entre les fabricants.

Concurrence, compatibilité et interopérabilité

La compatibilité d'un système d'exploitation est sa capacité à être utilisé à la place d'un autre, en particulier à exécuter les logiciels applicatifs de l'autre. Le système d'exploitation est dit « compatible » avec l'autre. La compatibilité au niveau source est la capacité pour un système d'exploitation A d'exécuter un logiciel applicatif créé pour B après avoir compilé le code source du logiciel pour la machine A. Et la compatibilité binaire est la capacité pour un système d'exploitation A d'exécuter un logiciel applicatif créé pour B tel quel, sans avoir à le recompiler.

L'interopérabilité est la capacité pour plusieurs systèmes à être utilisés ensemble, par exemple dans un même appareil, ou dans un réseau informatique.

Pour être compatibles, deux systèmes d'exploitation doivent avoir des points communs, notamment sur l'interface de programmation. La compatibilité binaire n'est possible qu'entre deux systèmes d'exploitation qui fonctionnent avec la même famille de processeur.

La compatibilité et l'interopérabilité entre les systèmes d'exploitation sont assurées, par les éditeurs, par alignement de leur produit sur des normes industrielles ainsi que des technologies rendues publiques.

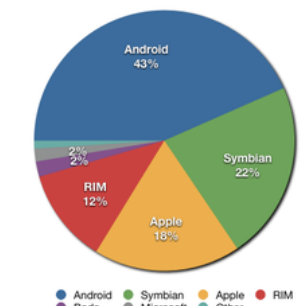
Le système d'exploitation Unix, créé en 1969, a servi de source d'inspiration pour toute une famille de systèmes d'exploitation. Le jeu de la concurrence, très vif dans les années 1980, a conduit les différents membres de la famille Unix à s'éloigner, et perdre la compatibilité les uns avec les autres. Des organismes de normalisation tels que Open Group se sont penchés sur le problème et ont édicté des normes garantissant la compatibilité à travers toute la famille Unix.

Entre 1995 et 2007, Microsoft, éditeur de la suite de systèmes d'exploitation Windows a été l'objet de plusieurs procès pour des pratiques anticoncurrentielles nuisant à la concurrence et à l'interopérabilité. La société a été condamnée par le département de la justice des États-Unis pour violation du Sherman Antitrust Act ainsi que par la Commission européenne pour violation des traités relatifs à la concurrence dans l'Union européenne.

La popularisation d'Internet dans les années 1990 a contribué à améliorer l'interopérabilité entre les systèmes d'exploitation.

La guerre des Unix et l'Open Group

Le système d'exploitation Unix a été développé par *American Telephone & Telegraph* (AT&T). Jusqu'en 1975, un accord avec l'État fédéral américain lui interdisait de commercialiser Unix, ayant le monopole de la téléphonie aux États-Unis, le code source du système d'exploitation était par conséquent public. Mais en 1975, ce monopole fut attaqué en justice et en 1982, la société était démembrée par décision de justice. AT&T, née de la fragmentation des activités d'American Telephone & Telegraph, put enfin commercialiser Unix : les ingénieurs de la société partirent du code source de la



Parts des systèmes d'exploitation des smartphones en 2011 dans le monde.

version 7 (ouverte) pour aboutir à UNIX System V. Simultanément, d'autres éditeurs s'inspirèrent de la version 7 pour créer des systèmes Unix, notamment l'université de Berkeley, avec sa Berkeley Software Distribution⁵⁰ (BSD, 1979). Puis ceux-ci ont servi de source d'inspiration pour d'autres systèmes d'exploitation, et ainsi de suite. En 2009, la famille Unix comptait plus de 25 systèmes d'exploitation.

Le jeu de la concurrence a conduit chaque éditeur à ajouter ses propres améliorations et ses propres fonctionnalités à son système d'exploitation optimisé pour un matériel en particulier. Ceci a amené les différents membres de la famille Unix à s'éloigner et perdre la compatibilité l'un avec l'autre.

En 1987, dans le but de réunifier la famille Unix, AT&T conclut un accord avec Sun Microsystems (un des principaux éditeurs de système d'exploitation fondé sur BSD). Les autres éditeurs ne voyant pas cet accord d'un bon œil, créent la fondation Open Software Foundation (OSF). Dans un même temps, l'Open Group, un consortium de normalisation, publie des normes relatives aux systèmes d'exploitation de la famille Unix⁵¹. Les deux institutions sont aujourd'hui fusionnées.

POSIX est le nom donné aux normes IEEE 1003. Cette famille de normes appartenant à l'Open Group a été lancée en 1988 et concerne l'interface de programmation. La conformité d'un système d'exploitation à cette norme assure la compatibilité au niveau source. En 2009, de nombreux systèmes d'exploitations sont conformes à cette norme, y compris en dehors de la famille Unix⁵².

POSIX fait partie de la Single Unix Specification, une norme lancée en 1994 par l'Open Group, qui concerne les systèmes de fichiers, les logiciels utilitaires, ainsi que 1 742 fonctions de l'interface de programmation⁵³. Le nom « Unix » appartient à l'Open Group et tout système d'exploitation doit être conforme à la *Single Unix Specification* pour être reconnu comme faisant partie de la famille Unix⁵⁴.

Microsoft et la concurrence

En 1995, conformément au Sherman Antitrust Act — une loi des États-Unis pour la prévention de l'abus de position dominante le département de la justice des États-Unis interdit à Microsoft certaines de ses pratiques considérées comme nuisibles à la concurrence. Deux ans plus tard, un procès est ouvert pour non-respect des interdictions de 1995 : Microsoft obtient l'annulation du procès sur l'argument que « la justice n'est pas équipée pour juger du bien-fondé du design des produits de haute technologie (sic) »

Entre 1999 et 2001, une enquête est ouverte concernant la position de Microsoft. L'enquête, menée par les juges Thomas Jackson et Richard Posner, amène à la conclusion que Microsoft abuse de sa position de monopole pour pratiquer du « favoritisme » sur le marché connexe des navigateurs Web, des pratiques qui nuisent à ses concurrents, gênent leur émergence et freinent l'innovation⁵⁵. La société échappe de peu à la scission, et se retrouve dans l'obligation de publier les spécifications de ses technologies, en particulier les interfaces de programmation et les protocoles réseau, ceci afin de préserver l'interopérabilité et la concurrence⁵⁶.

Plus tard, en 2007, Microsoft est condamné par la Commission européenne à une amende de près de 500 millions d'euros pour violation de l'article 82 du traité CE et l'article 54 de l'accord EEE (textes relatifs au droit de la concurrence et l'abus de position dominante) après avoir refusé de publier une de ses spécifications techniques à son concurrent Sun Microsystems^{N 19}. Selon la Commission européenne, les agissements de Microsoft nuisent à l'interopérabilité des systèmes d'exploitation et à la concurrence⁵⁷.

Les réseaux informatiques

Chaque système d'exploitation comporte une palette de programmes relatifs aux protocoles réseau. La composition de la palette dépend du choix de l'éditeur et diffère d'un système d'exploitation à l'autre. Toutefois, deux ordinateurs ne peuvent communiquer ensemble qu'à la condition unique d'utiliser les mêmes protocoles.

La popularisation d'Internet dans les années 1990 a poussé de nombreux éditeurs à inclure dans leur système d'exploitation des programmes relatifs aux protocoles TCP/IP (les protocoles d'Internet), améliorant ainsi l'interopérabilité entre les systèmes d'exploitation.

Notes et références

Notes

- 1. Plus précisément, ni le code, ni les données, ni la pile n'ont besoin d'être totalement en mémoire centrale.
- 2. Le fait de transférer un segment complet en mémoire de masse est nommé *swap out*, l'opération inverse *swap in*.
- 3. D'abord sous forme de téléscripteurs, puis de terminaux passifs
- 4. Le langage C est un langage de programmation souvent utilisé pour les systèmes d'exploitation en raison de sa puissance et du contrôle total qu'a le programmeur sur la mémoire (A. Tanenbaum, *Systèmes d'exploitation* 1.8).
- 5. Ce mécanisme qui permet au système d'interrompre un programme en cours d'exécution est nommé *préemption* (A. Tanenbaum, *ibid.*, 2.4.1, p. 140).
- 6. La totalité du code source se trouve dans son livre *Operating Systems : Design and Implementation*
- 7. L'élimination des limitations de Minix aurait rendu le code source trop difficile à lire et à comprendre par les étudiants en un semestre (A. Tanenbaum, *ibid.*, 10.1.7, p. 766).
- 8. Andrew Tanenbaum, enseignant, utilisait le code source de Minix pour illustrer ses cours sur les systèmes d'exploitation.
- 9. anciennement appelé EPOC.
- 10. racheté à Psion.
- 11. Compatible PC, Macintosh, DEC Alpha, Sparc, Itanium, m68k, etc.
- 12. Via la branche du noyau Linux-rt.
- 13. Entre 1986 et 1991, les produits de la famille Windows étaient des environnements graphiques pour le système d'exploitation DOS.
- 14. Scission du projet 386BSD.
- 15. Plus de 50 types de machines. Cf. (en) Liste des plateformes compatibles avec NetBSD(http://www.netbsd.org/ports/)
- 16. Compatible PC, ARM, MIPS, Macintosh, DEC Alpha, Sparc, Itanium, Sun Microsystems, Xbox.
- 17. Compatible PC, ARM, DEC Alpha, Sparc, etc.
- 18. Créé par fusion entre NeXTSTEP et Mac OS.
- 19. L'échange de spécifications est une pratique courante dans le marché informatique.

Références

- 1. (en) Brian L. Stuart, *Principles of Operating Systems: Design & Applications*, Cengage Learning EMEA 2008 (ISBN 978-1418837693).
- 2. (en) I. A. Dhotre, *Operating Systems*, Technical Publications 2009 (ISBN 978-8-1843-1644-9).
- 3. (en) Achyut S. Godbole et Atul Kahate, *Operating Systems*, 3^e ed., Tata McGraw-Hill Education, 2011 (ISBN 978-0-0707-0203-9)
- 4. (en) Top 5 Operating Systems from July 2008 to April 2012 (http://gs.statcounter.com/#os-ww-monthly-200807-201204) - StatCounter Global Stats
- 5. (en) Pabitra Pal Choudhury *Operating Systems: Principles and Design*, Prentice-Hall of India Pvt. Ltd, 2009 (ISBN 978-8-1203-3811-1).
- 6. (en) Sibsanekar Halder et Alex A. Aravind *Operating Systems*, Pearson Education India - 2010, (ISBN 9788131730225)
- 7. (en) Jose Garrido, Richard Schlesinger et Kenneth Hoganson *Principles of Modern Operating Systems*, Jones & Bartlett Publishers - 2011, (ISBN 9781449626341)
- 8. (en) *Operating Systems*, Technical Publications, (ISBN 9788184315622)
- 9. (en) Balagurusamy *Fund Of Computers*, Tata McGraw-Hill Education, (ISBN 9780070141605)
- 10. (en) *Operating Systems: Principles And Design*, PHI Learning Pvt. Ltd. - 2009, (ISBN 9788120338111)

11. (en) George A. Anzinger et Adele M. Gadol, « A Real-Time Operating System with Multi-Terminals and Batch/Spool capabilities », *HP journal*, 1975 (lire en ligne (<http://www.hpmuseum.net/document.php?catfile=349>))**[PDF]**.
12. (en) Pramod Chandra P Bhatt, *An Introduction To Operating Systems: Concepts and Practice*, 2^d Ed., Prentice-Hall of India Pvt. Ltd, 2007,(ISBN 9788120332584)
13. (en) Amjad Umar, *Third Generation Distributed Computing Environments*NGE Solutions Inc, 2004,(ISBN 9780975918210)
14. (en) Godbole, *OPERATING SYSTEM 3E*, Tata McGraw-Hill Education, 2011,(ISBN 9780070702035)
15. « MIT - General Electric, Honeywell-Bull - MULTICS » (<http://www.feb-patrimoine.com/PROJET/multics/multics.htm>), sur *feb-patrimoine.com* (consulté le 30 septembre 2013)
16. (en) Ayan Moumina, « History of operating systems » (<http://www.bhu.ac.in/ComputerScience/viv&os/1f-Moumina.pdf>) **[PDF]** (consulté le 30 septembre 2013).
17. (en) Mark Bellis, « Putting Microsoft on the Map - History of the MS-DOS Operating Systems, IBM & Microsoft » (<http://inventors.about.com/library/weekly/aa033099.htm>) sur *about.com* (consulté le 30 septembre 2013).
18. (en) « History of the graphical user interface » (<http://www.sensomatic.com/chz/gui/history.html>), sur *sensomatic.com* (consulté le 30 septembre 2013).
19. « L'annonce originale du projet GNU » (<http://www.gnu.org/gnu/initial-announcement.fr.html>), sur *gnu.org*, 27 septembre 1983 (consulté le 30 septembre 2013)
20. (en) Richard Stallman, « The Hurd and Linux » (<http://www.gnu.org/software/hurd/hurd-andlinux.html>) (consulté le 30 septembre 2013)
21. (en) Linus Torvalds, « RELEASE NOTES FOR LINUX v0.12 » (<http://www.kernel.org/pub/linux/kernel/Historic/old-versions/RELNOTES-0.12>) The Linux Kernel Archives, 5 janvier 1992 (consulté le 30 septembre 2013)
22. (en) Josh Schneider, « Finding aid to the Berkeley Software Distribution » (<http://www.oac.cdlib.org/data/13030/b5/hb2b69n7b5/files/hb2b69n7b5.pdf>) **[PDF]** (consulté le 30 septembre 2013), p. 3.
23. (en) Elizabeth U. Harding, « Unix pioneer ends BSD research », *Software Magazine*, octobre 1992 (lire en ligne (<http://www.anonymous-insidernet/unix/research/1992/10.html>))
24. (en) Working group on Libre Software, *Free Software / Open Source: Information Society Opportunities for Europe*, avril 2000 (lire en ligne (http://eu.conecta.it/paper/brief_history_open_source.html)), « A brief history of open open source software »
25. (en) Michael Kifer, Scott Smolka, *Introduction to Operating System Design and Implementation: The OSP 2 Approach*, Springer Science & Business Media - 2007, (ISBN 9781846288432)
26. (en) *Operating Systems: Principles And Design* PHI Learning Pvt. Ltd., 2009, (ISBN 9788120338111)
27. (en) Er. Vivek Sharma - Er Manish Varshney et Shantanu Sharma, *Design and Implementation of Operating System*, Laxmi Publications, Ltd. - 2010, (ISBN 9789380386416)
28. (en) Brian L. Stuart, *Principles of Operating Systems: Design & Application*, Cengage Learning EMEA - 2008, (ISBN 9781418837693)
29. « Théorie des systèmes d'exploitation : Introduction » (<http://doc.esaia.org/~ptregouet/Cours/TSE/TSE%201%20Introduction.pdf>) sur <http://doc.esaia.org> (consulté le 18 janvier 2018), p. 7
30. (en) « Amiga Operating System Frequently Asked Questions » (<http://www.cinereal.com/computers/faqamigos.shtml>), sur *CineReal*, 2005 (consulté le 30 septembre 2013)
31. (en) « IBM AIX Systems management - Desktop » (<http://www-03.ibm.com/systems/power/software/aix/sysmgmt/desktop/index.html>) IBM (consulté le 30 septembre 2013)
32. (en) Andy Tanenbaum, « Introduction to Minix 3 » (http://www.osnews.com/story/15960/Introduction_to_MINIX_3) OSNews, 25 septembre 2006 (consulté le 30 septembre 2013)
33. (en) John Pagonis, « Symbian OS Presentation » (<http://www.pagonis.org/Presentations/EE27.pdf>) **[PDF]**, 2 mars 2004 (consulté le 30 septembre 2013).
34. (en) « Kernel.exe Describes the Kernel and VxD for Windows 95 » (<http://support.microsoft.com/kb/181300/en-us>) sur *support.microsoft.com* (consulté le 30 septembre 2013)
35. (en) « Windows CE and Windows Mobile Timeline » (<http://www.networkdictionary.com/Software/Windows-CE.php>) Archive (<http://web.archive.org/web/http://www.networkdictionary.com/Software/Windows-CE.php>) • Wikix (<http://archive.wikix.com/cache?url=http://www.networkdictionary.com/Software/Windows-CE.php>) • Archive.is (<http://archive.is/http://www.networkdictionary.com/Software/Windows-CE.php>) • Google (<https://www.google.fr/search?q=cache:http://www.networkdictionary.com/Software/Windows-CE.php>) • Que faire ?
36. (en) « Windows CE Kernel services » (<http://msdn.microsoft.com/en-us/library/ms906442.aspx>) Microsoft Developer Network (consulté le 30 septembre 2013)
37. « http://www.lambarque.com/extension-tempsreel-rtx-pour-windows-passe-au-64-bits_000132 »
38. « Chiffres clés : les systèmes d'exploitation sur PC » (<http://www.zdnet.fr/actualites/chiffres-cles-les-systemes-d-exploitation-sur-pc-39790131.htm>)
39. (en) « Windows market share drop to 15 years low » (<http://www.tgdaily.com/content/view/full/40398/113/>) TG Daily (consulté le 30 septembre 2013)
40. (en) Sebastian Anthony, « Microsoft's share of the consumer market has dropped from 95% to 20% in 8 years » (<http://www.extremetech.com/computing/14277-microsofts-share-of-the-consumer-market-has-dropped-from-95-to-20-in-8-years>) Extreme Tech (consulté le 30 septembre 2013)
41. (en) « Unix Flavor List » (<http://linux.about.com/library/bl/flavorlist.htm>) sur *about.com* (consulté le 30 septembre 2013).
42. (en) Jeff Drew, « IDC report: IBM widens lead as server market shrinks again » (<http://triangle.bizjournals.com/triangle/stories/2009/08/31/daily35.html>) Triangle Business Journal, 2 septembre 2009 (consulté le 30 septembre 2013).
43. (en) « Usage of operating systems for websites » (http://w3techs.com/technologies/overview/operating_system/all) W3Techs - Web Technology Surveys, septembre 2013 (consulté le 30 septembre 2013).
44. (en) « The triumph of Linux as supercomputer OS » (<http://royal.pingdom.com/2009/06/24/the-triumph-of-linux-as-a-supercomputer-os/>) sur *Royal Pingdom* (consulté le 30 septembre 2013).
45. « Canals, 115 millions de smartphones en 2007 » (<http://www.generation-nt.com/canalys-perspectives-smartphones-2007-actualite-67815.html>) GNT - portail des nouvelles technologies et du jeu (consulté le 30 septembre 2013) : « Sur l'ensemble de l'année, le système d'exploitation Symbian occupe 67 % de parts de marché, Windows Mobile 13 % et RIM 10 % ».
46. (en) Emil Protalinski, « Android grabs 75% market share in Q3, followed by 14,9% for iOS and 4,3% for BlackBerry » (<http://thenextweb.com/mobile/2012/11/01/android-grabs-75-0-market-share-in-q3-followed-by-14-9-for-ios-and-4-3-for-blackberry/>) TheNextWeb (consulté le 30 septembre 2013).
47. « Tablettes tactiles : 83,9 % de part de marché pour Apple iOS » (<http://www.journaldunet.com/solutions/systemes-reseaux/marche-os-tablettes-tactiles-2010-2015.shtml>) Journal du Net, 11 avril 2011 (consulté le 30 septembre 2013).
48. (en) « Worldwide Tablet Market Surges Ahead on Strong First Quarter Sales, Says IDC » (<http://www.idc.com/getdoc.jsp?containerId=prUS24093213>) International Data Corporation, 1^{er} mai 2013 (consulté le 30 septembre 2013).
49. (en) « Linux Rules Supercomputers » (https://www.forbes.com/2005/03/15/cz_dl_0315linux.html), sur *forbes.com*, 15 mars 2005 (consulté le 30 septembre 2013) : « Linux now has become so technically powerful that it lays claim to a prestigious title—it runs more of the world's supercomputers »
50. Cf. (en) Michael Kerrisk, *The LINUX Programming Interface* San Francisco, No starch Press, 2010, 1508 p. (ISBN 9781593272203), « History and Standards » p. 3.
51. (en) « Unix history and timeline » (http://www.unix.org/what_is_unix/history_timeline.html), sur *unix.org* (consulté le 30 septembre 2013).
52. (en) « POSIX.1 FAQ » (http://www.opengroup.org/austin/papers/posix_faq.html), sur *opengroup.org*, 5 octobre 2011 (consulté le 30 septembre 2013).
53. (en) « Single Unix specification FAQ » (http://www.opengroup.org/austin/papers/single_unix_faq.html) sur *opengroup.org*, 5 octobre 2011 (consulté le 30 septembre 2013).
54. (en) « What is UNIX » (http://www.unix.org/what_is_unix.html) sur *opengroup.org* (consulté le 30 septembre 2013).
55. (en) Jeffrey August Eisenach et Thomas M. Lenard, *Competition, innovation, and the Microsoft monopoly* Springer, 1999, (ISBN 9780792384649)
56. (en) Nicholas Economides, « Microsoft Antitrust, a case study » (http://www.stern.nyu.edu/networks/homeworks/Microsoft_Case.pdf) **[PDF]**, avril 2003 (consulté le 30 septembre 2013).
57. « Procédure d'application de l'article 82 du traité CE et de l'article 54 de l'accord EEE engagée contre Microsoft Corporation » (<http://eur-lex.europa.eu/LexUriServ/LexUriServlet.do?uri=CELEX:32007D0053:FR:HTML>) sur *EUR-Lex.com*, 24 mai 2004 (consulté le 30 septembre 2013).

A. Tanenbaum, Systèmes d'exploitation

1. 3.3, p. 196.
2. 3.3.1, p. 196-200.
3. 3.3.3, p. 203-204.
4. 3.6.1, p. 236.
5. Algorithmes de remplacement de pages, ch. 3.4p. 209).
6. 1.2.3, p. 15.
7. 10.2.1, p. 760.
8. 10.2.1, p. 761 et p. 762.
9. 1.2.4, p. 17-18.
10. 10.1.7, p. 766.
11. 10.1.7, p. 768

Bibliographie


- Andrew Tanenbaum, *Systèmes d'exploitation* Pearson, 2008, 3^e éd. [détail de l'édition]
- Laurent Bloch, *Les Systèmes d'exploitation des ordinateurs. Histoire, fonctionnement, enjeux* Vuibert, 2003 (ISBN 978-2-7117-5322-2) [lire en ligne]

Annexes

Articles connexes

- Ordonnancement
- Pilote informatique
- Interruption
- Système de fichiers
- Environnement de bureau
- Noyau de système d'exploitation
- Liste des noyaux de systèmes d'exploitation
- Liste des systèmes d'exploitation
- Système d'exploitation temps réel
- Catégorie:Système d'exploitation
- Système d'exploitation distribué

Sur les autres projets Wikimedia :

 *Système d'exploitation* sur Wikimedia Commons

 *système d'exploitation* sur le Wiktionnaire

 *Système d'exploitation* sur Wikiversity

 *Système d'exploitation* sur Wikibooks

Liens externes

- Notion sur les systèmes d'exploitation- Comment ça marche
- L'ascension des systèmes d'exploitation Microsoft - Génération-NT, 29 novembre 2005
- Historique des systèmes d'exploitation, et des réseaux micro-informatique sur PC
- **(en)** La Bible Internet des systèmes d'exploitation principes, historique, familles, spécifications…

Ce document provient de «https://fr.wikipedia.org/w/index.php?title=Système_d%27exploitation&oldid=147156412».

La dernière modification de cette page a été faite le 5 avril 2018 à 23:05.

Droit d'auteur : les textes sont disponibles sous licence Creative Commons attribution, partage dans les mêmes conditions ; d'autres conditions peuvent s'appliquer. Voyez les conditions d'utilisation pour plus de détails, ainsi que les crédits graphiques. En cas de réutilisation des textes de cette page, voyez comment citer les auteurs et mentionner la licence. Wikipedia® est une marque déposée de la Wikimedia Foundation, Inc, organisation de bienfaisance régie par le paragraphe 501(c)(3) du code fiscal des États-Unis.